The Serial Stommel Model in FORTRAN

Luke Lonergan
High Performance Technologies, Inc.

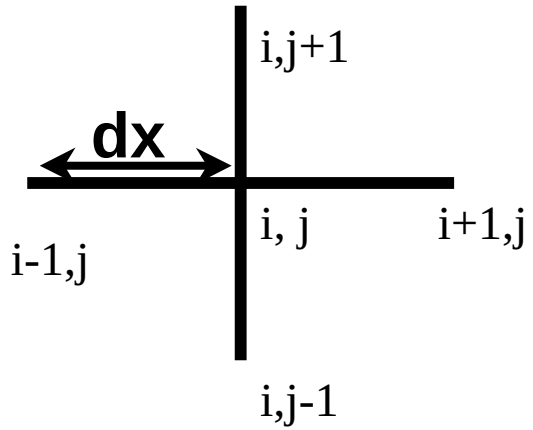
Prototype Model Overview

After this exercise, we will have a well understood prototype model implemented in FORTRAN
In the last section we learned that the Stommel model is a linear partial differential equation with constant coefficients:

$$\gamma\nabla^2\psi + \beta\frac{\partial\psi}{\partial x} = \alpha\sin\left(\frac{\pi y}{L_y}\right)$$

Discretization of the Equations

We also defined a computational grid of points in (x,y) and (i,j) coordinates:



interior grid points:
i=2,nx-1 ; j=2,ny-1

boundary points:

(i,1) & (i,ny) ; i=1,nx

(1,j) & (nx,j) ; j=1,ny

Next step is to solve the equation approximately using Fixed Point Iteration

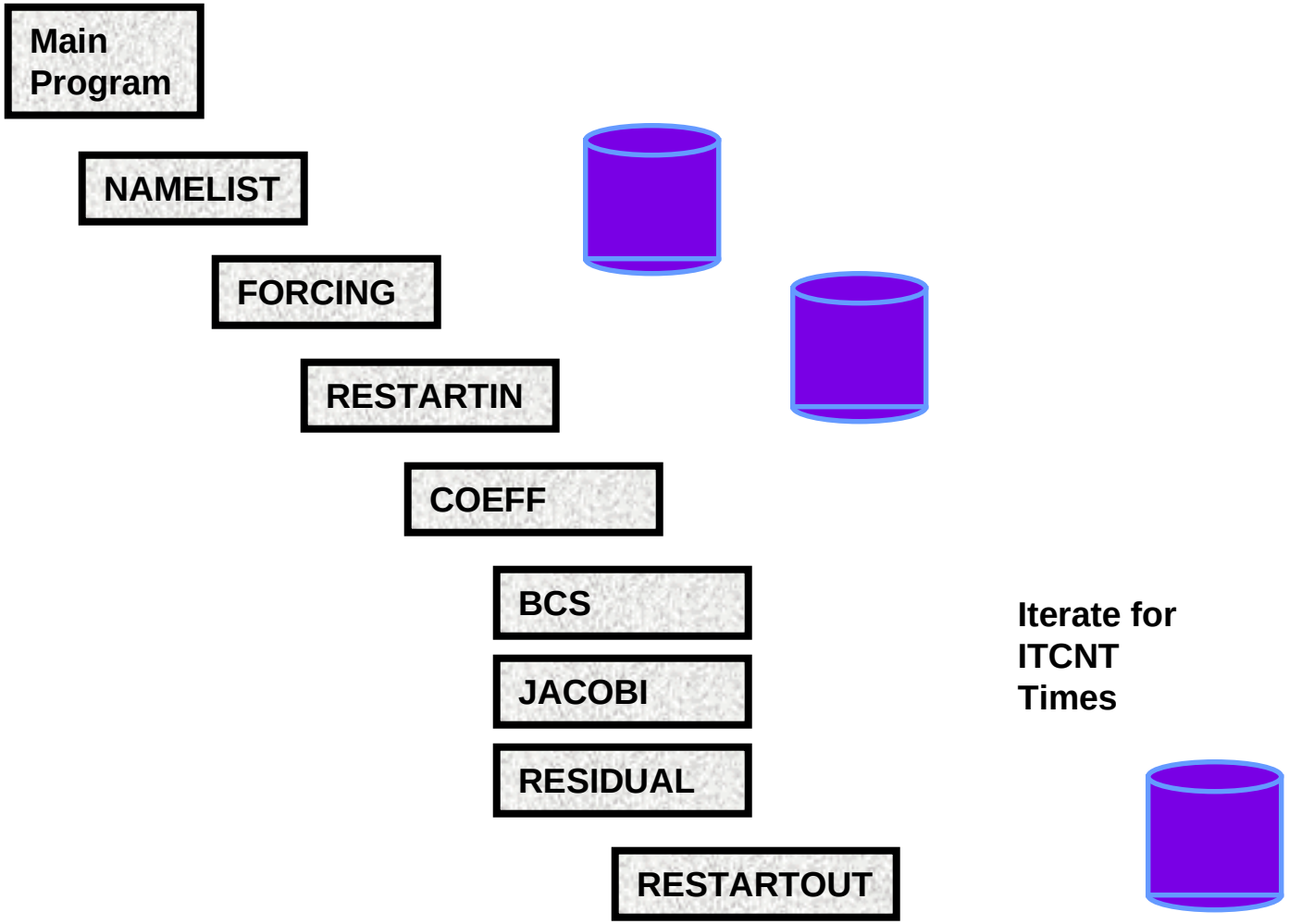
Iterative Scheme

The iterative equation is written:

$$\psi_{i,j}^{n+1} = a_1 \psi_{i+1,j}^{n} + a_2 \psi_{i-1,j}^{n} + a_3 \psi_{i,j+1}^{n} + a_4 \psi_{i,j-1}^{n} - a_5 f_{i,j}$$

Where the (n) denotes the “old” value and (n+1) is the “new” value. Note that the forcing function doesn’t change with each iteration, neither do the constants a_1 through a_5

Flowchart for the Serial Code



Develop the Basic Code Variables

Change directory to serial/step1
Edit the README file
Recommended Approach:
    Review the finished code example in serial/step1/completed
Advanced:
    Make any style changes and copy modified code to serial/step2
Duration: 5 Minutes

Develop the Basic Code Structure

Change directory to serial/step2
Edit the README file
Recommended Approach:
    Review the finished code example in serial/step2/completed
Advanced:
    Make any style changes and copy modified code to serial/step3
Duration: 5 Minutes

Develop A Runnable Code

Change directory to serial/step3

Edit the README file

Recommended Approach:

- Review the finished code example in serial/step3/completed
- Compile and link the code using
  - “make”
- Execute the code, perhaps changing the dimensions and iteration count
- Time the code using
  - “time executable_name”

Basic Profile of the Code

Change directory to serial/step4
Edit the README file
Recommended Approach:

- Review the finished code example in serial/step4/completed
- Compile and link the code using
  - “make”
- Execute the code and take note of the performance
- Time the separate parts of the program, noting the relative contributions

Runnable Code (Cont'd)

Advanced:

Make any style changes and copy modified code to serial/step4

Duration: 10 Minutes

Basic Profiling (Cont'd)

Advanced:

Make any style changes and copy modified code to serial/step5

Duration: 10 Minutes

Insertion of Basic Input/Ouput

Change directory to serial/step5

Edit the README file

Recommended Approach:

- Review the finished code example in serial/step5/completed
- Compile and link the code using
  - “make”
- Execute the code and review the functionality of the I/O
- Experiment with restarting the code from the previous run

Basic I/O (Cont'd)

Example:

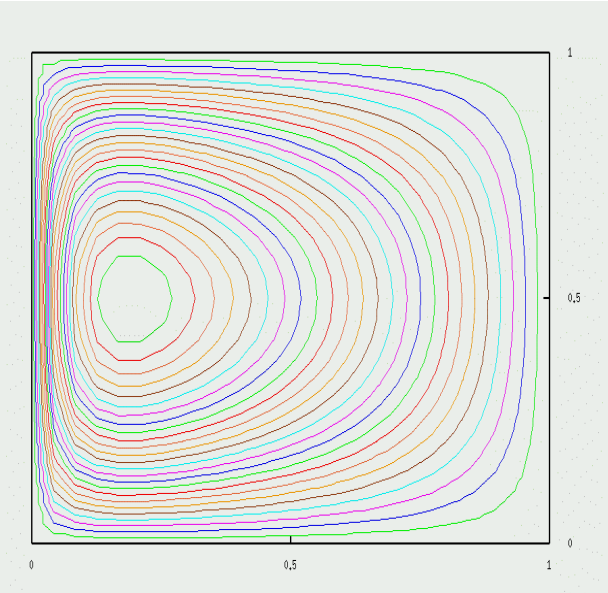
```
stommel.x
mv psi.out psi.in
stommel.x
(etc.)
```

Try plotting the resulting solution using the "plotter.x" program and GNUPlot

Example:

```
plotter.x
(enter the name of the solution file, e.g.
"psi.out")
(Follow the directions for GNUPlot)
```

Basic I/O (Cont’d)



Advanced:

Make any style changes and copy modified code to serial/step6

Duration: 15 Minutes

Measure Performance /Optimize

Change directory to serial/step6

Edit the README file

Recommended Approach:

- Review the finished code example in serial/step6/completed
- Compile and link the code using
  - “make”
- Execute the code and review the performance baseline. Write down the first number for Mflops you get and

Measure Performance /Optimize

try to improve it!

Note that there are a series of suggested approaches in the README file. In order to use the completed example code:

- edit the Makefile and replace main1.o with main2.o
- recompile (type “make”)
- execute the program
- repeat with main2.o replaced with main3.o

Advanced:

- Try running on different systems!

Duration: 15 Minutes